

UNIVERSIDAD CARLOS III DE MADRID
AREA DE ARQUITECTURA Y TECNOLOGÍA DE COMPUTADORES
GRADO EN INGENIERÍA INFORMÁTICA. SISTEMAS DISTRIBUIDOS

Para la realización del presente examen se dispondrá de **3 horas**. **NO** se podrán utilizar libros, apuntes ni calculadoras de ningún tipo.

Alumno: _____ Grupo: _____

Ejercicio 1 (2,5 puntos). Responda a las siguientes preguntas cortas justificando brevemente su respuesta:

- a) Dado el siguiente mensaje SOAP: ¿Qué protocolo de aplicación usa? Identifique los campos principales del mensaje y describa brevemente cuál es su contenido.

```
POST / engelen/calserver.cgi HTTP/1.1
Host: webserv.cs.fsu.edu
User-Agent: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 464
Connection: close
SOAPAction: ""

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:c="urn:calc">
  <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
    <c:add>
      <a>1</a>
      <b>2</b>
    </c:add>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

- b) Dado el siguiente sistema distribuido compuesto por $N=7$ nodos. Aplique el método de votación (quórum) para:
- Determinar la combinación de nodos R (copias de lectura) y nodos W (copias de escritura) que formarían un quórum válido.
 - Si el coste de lectura es la mitad del coste de escritura y la probabilidad de lectura $p=0.35$, ¿qué combinación de R y W de las anteriores sería la más eficiente? Justifique su respuesta.

- c) Dado el siguiente sistema distribuido compuesto por N nodos, N=6, que usa votación dinámica para mantener la consistencia de las réplicas. Los nodos tienen los siguientes valores NV y SC para una determinada réplica (NV: número de versión y SC: cardinalidad de la actualización):

	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5	Nodo 6
NV	5	5	5	6	6	6
SC	6	6	6	3	3	3

- ¿Se puede actualizar la réplica en la partición {4,5,6}? Rellene apropiadamente la siguiente tabla y justifique la respuesta.

	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5	Nodo 6
NV						
SC						

- Una fallo de red fragmenta la partición {4,5,6} en dos: {4} {5,6}. En esta situación se puede actualizar la réplica en la partición {4}? Justifique su respuesta.

- d) En un cliente NFS, un programa de usuario realiza las siguientes operaciones sobre un fichero:

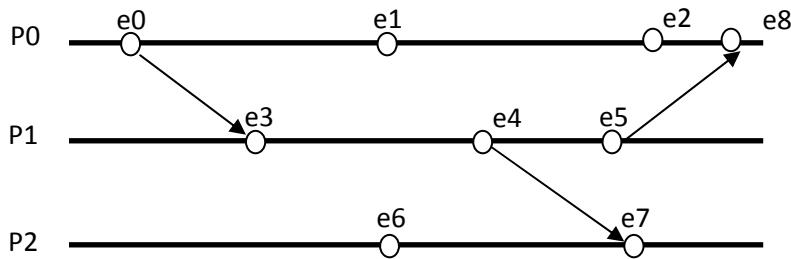
1. `df=open("/mnt/home/usuario1/foo.txt", O_RDWR)`
2. `read(df,buffer,1024)`
3. `replace(buffer,buffer1,100)`
4. `write(df,buffer1,1024)`
5. `close(df)`

NOTA: Asuma que en el cliente se ejecutó la orden: `mnt -t nfs /home/ /mnt`

- Describir el conjunto de llamadas NFS necesarias para realizar esas operaciones y qué argumentos necesita cada una de estas llamadas.
- NFS es un servidor sin estado. ¿Qué quiere decir? ¿Cómo afecta esto a los datos que envía el cliente al servidor?

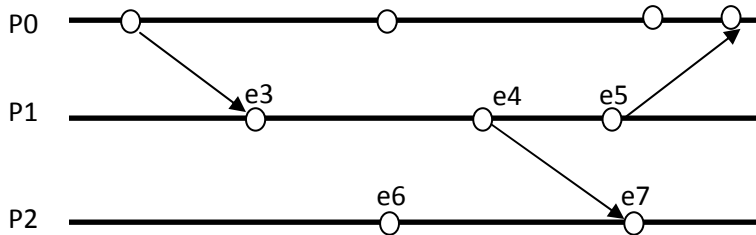
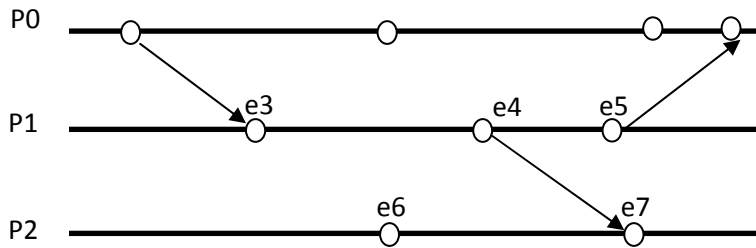
- e) ¿En qué consiste la computación voluntaria? Cite algunos ejemplos de aplicaciones.

Ejercicio 2 (1,5 puntos). Dados los siguientes procesos P1, P2 y P3 que se encuentran ejecutando en un sistema distribuido, y que producen los eventos mostrados en la siguiente figura.



Se pide:

- Definir las relaciones de causalidad de Lamport entre los eventos que aparecen en la figura.
- Definir para qué eventos no es posible establecer las relaciones de causalidad de Lamport. Justifique su respuesta.
- Usando los relojes lógicos de Lamport, indique las marcas de tiempo para los eventos de los procesos anteriores.



- Si $RL1(E0) < RL2(E3)$, ¿sería posible decir que $e0$ precede a $e3$? Justifique su respuesta
- ¿Cuál es el orden que aseguran las marcas de tiempo de Lamport? Justifique su respuesta.

Ejercicio 3 (2'5 puntos). Se desea implementar el algoritmo de exclusión mutua distribuida basado en coordinador usando paso de mensajes. En este algoritmo, uno de los nodos actúa como coordinador. Cuando el proceso **i** (diferente al coordinador) quiere entrar en la sección crítica envía un mensaje al coordinador:

send_entrada(i):

La función del coordinador es decidir si el proceso que solicitante puede o no entrar en la sección crítica. Si en el momento de recibir el mensaje ningún otro proceso está ejecutando la sección crítica, el coordinador permitirá al solicitante entrar y para ello le enviará el mensaje:

send_ok(i)

Si por el contrario, hay algún otro proceso ejecutando en la sección crítica, el coordinador no responderá al proceso solicitante hasta que la sección crítica quede libre. Cuando un proceso **i** sale de la sección crítica, enviará el mensaje al coordinador indicando que abandona la sección crítica:

send_salida (i)

Se pide:

- a) Diseñe un conjunto de primitivas de recepción válidas para poder recibir los mensajes intercambiados entre un proceso **i** y el coordinador.
- b) Implemente el código del proceso coordinador utilizando las primitivas especificadas en este enunciado.

Ejercicio 4 (3,5 puntos). Una empresa de juegos on-line pretende implementar una versión básica del juego “Apalabrados”. En este juego, un usuario compone palabras a partir de otras palabras formadas por otros usuarios y una serie de letras aleatorias. Para que el usuario pueda empezar a jugar debe primero registrarse y posteriormente iniciar una partida con otro usuario ya registrado. Una vez iniciada la partida, los usuarios componen palabras y las envían al servidor para su validación. Si la palabra es correcta, el servidor calculará su puntuación y devolverá al usuario dicha puntuación. El objetivo del juego es obtener más puntos que el rival. La partida termina cuando decide el usuario o cuando un usuario gana la partida (se acaban todas las letras y tiene máxima puntuación).

Se desea implementar un sistema distribuido que proporcione el servicio “Apalabrados”. Los servicios básicos que se deben ofrecer son los siguientes:

- 1) Registro del usuario: un usuario se registra en el sistema con sus datos personales.
- 2) Iniciar una partida: un usuario inicia una nueva partida con otro usuario.
- 3) Enviar una palabra: un usuario envía una palabra en una partida empezada para poder puntuar en dicha partida.
- 4) Terminar una partida: un usuario decide abandonar una partida previamente iniciada.

Se pide:

- a) Diseñar la aplicación cliente-servidor anterior utilizando sockets, indicando y especificando todos los aspectos necesarios para su diseño. Como parte del diseño, describa detalladamente el protocolo de servicio.
- b) De acuerdo al diseño anterior, indique qué llamadas a la biblioteca de sockets utilizaría en el cliente y en el servidor y en qué orden.
NOTA: Indique al menos los argumentos más relevantes de las funciones de sockets.
- c) Considerando que se emplean las RPC de Sun, defina la interfaz necesaria para poder implementar la aplicación cliente-servidor anterior.